

Ihnen fallen hierfür sicher viele interessante Anwendungen ein. PDF-Unterstützung gibt es für die meisten Programmiersprachen von Bedeutung.



HACK

#86

## Russische Puppen

PDF-Dokumente in andere PDF-Dokumente integrieren

Grafiken können Sie mit pdf $\LaTeX$  ganz so einbinden wie mit gewöhnlichem  $\LaTeX$  (siehe *Ein Bild sagt mehr als tausend Wörter* [Hack #74]). Die gängige Methode mit dem Kommando `\includegraphics` eignet sich aber nicht für »formatfüllendes« Material, etwa wenn Sie eine komplette DIN-A4-Seite aus einem anderen Dokument in Ihr DIN-A4 großes Dokument einbauen wollen. (Stellen Sie sich zum Beispiel vor, Sie setzen die Zeitschrift Ihres Kleintierzuchtvereins und müssen die ganzseitige Anzeige einer Wellensittichfutter-Firma unterbringen.)

Auch dabei lässt pdf $\LaTeX$  Sie zum Glück nicht im Stich: Das `pdfpages`-Paket von Andreas Matthias ist genau für diese Anwendung gedacht. Im einfachsten Fall können Sie mit etwas wie

```
\usepackage{pdfpages}
```

```
...
```

```
ist dieses Zwergkaninchen ein echter Riese.
```

```
\includepdf{anzeige}
```

```
Der Goldfisch dagegen ist im Vergleich zum Kaninchen ein ...
```

```
...
```

die erste Seite von *anzeige.pdf* an genau dieser Stelle in Ihr Dokument integrieren. Sollte die aktuelle Seite mit der Anmerkung über das Karnickel gerade halb voll sein, dann wird sie ausgeworfen, die Anzeige erscheint auf der nächsten Seite, und mit dem Goldfisch geht es auf der Seite danach weiter.

Sollten Sie diese Art von lästiger Unterbrechung nicht schätzen, hilft Ihnen zum Beispiel das `afterpage`-Paket von David Carlisle. Es dient dazu, Befehle voranzumelden, die ausgeführt werden, wenn  $\LaTeX$  die aktuelle Seite ausgegeben hat und bevor es die nächste anfängt. (Eine typische Anwendung von `afterpage` besteht darin,  $\LaTeX$  auf die Sprünge zu helfen, wenn Ihre gleitenden Abbildungen und Tabellen nicht so gleiten, wie sie sollen, sondern sich aufstauen und am Dokumentende als großer Klumpen ausgegeben werden: Ein `\afterpage{\clearpage}` sorgt dafür, dass nach der Seite, auf der das Kommando steht, alle anstehenden Tabellen und Abbildungen ausgegeben werden. Danach geht es dann wieder von vorne los.)

Dies ist ein Auszug aus dem Buch *Linux in TeX*, ISBN 978-3-89721-471-0  
<http://www.oreilly.de/catalog/latexhksger/>  
 Dieser Auszug unterliegt dem Urheberrecht. © O'Reilly Verlag 2007

In unserem Fall erreichen wir über

```
\afterpage{\includepdf{anzeige.pdf}}
```

dass die aktuelle Seite noch komplett gefüllt wird, dann die Anzeige erscheint und es dann mit dem Artikel weitergeht, als sei nichts gewesen. Sie müssen nur darauf achten, das `\afterpage`-Kommando so zu platzieren, dass es auf der Seite *vor* derjenigen steht, wo die Anzeige gewünscht war – jedenfalls wenn eine spezielle Seite für das eingebundene Material vorgesehen ist. (Die Heftmitte ist zum Beispiel populär für eigens produzierte Extras. Denken Sie zum Beispiel an gewisse Magazine mit leicht geschürzten Damen.  $\LaTeX$  ermöglicht es Ihnen bisher leider nicht, gezielt etwa `\includepdf` für die Seiten 15 und 16 eines 32-seitigen Heftchens zu sagen – Sie müssen die Seite 14 schon selber finden –, aber völlig unmöglich wäre es nicht.)

## Mehrere Seiten einbinden

Standardmäßig holt `\includepdf` immer nur die erste Seite eines Dokuments, egal wie viele Seiten das Dokument in Wirklichkeit hat. Sie können aber gezielt mehrere Seiten einbinden, indem Sie die `pages`-Option verwenden.

```
\includepdf[pages=1-5]{subdocument}
```

holt zum Beispiel die Seiten 1 bis 5 aus *subdocument.pdf*.

```
\includepdf[pages=3-]{subdocument}
```

holt alle Seiten ab Seite 3 bis zum Dokument-Ende,

```
\includepdf[pages=1,3,7]{subdocument}
```

nur die Seiten 1, 3 und 7, und

```
\includepdf[pages=-]{subdocument}
```

holt alle Seiten des Dokuments. Außerdem können Sie zusätzliche leere Seiten einbinden, indem Sie statt einer Seitennummer ein »{}« in die `pages`-Liste aufnehmen. (Das ist natürlich weniger interessant, wenn Sie PDF-Seiten in ein anderes Dokument einbauen wollen – `pdfpages` bildet aber auch die Basis des Programms `pdfnup` [Hack #33], und dort ist diese Funktion sehr nützlich.)



HACK  
#87

## Vom Scanner zur PDF-Datei

Eine PDF-Datei aus einzelnen Seiten zusammensetzen – mit Inhaltsverzeichnis

Wie kommt man von einem Verzeichnis mit einzelnen gescannten Seiten zu einem PDF-Dokument, das vielleicht noch passende Verweise auf die einzelnen